

# Zárójelentés

OTKA nyilvántartási szám: T 042459

“Számítógépes alkalmazások teljesítményanalízise és optimalizációja”

Témavezető: Dr. Kacsuk Péter

Magyar Tudományos Akadémia  
Számítástechnikai és Automatizálási Kutató Intézete

2006

### A kutatási téma szakmai zárójelentése

A számítási hálózat (computational grid) olyan szoftver és hardver infrastruktúra, amely földrajzilag elosztott, nagyszámú, heterogén erőforrások megosztását teszi lehetővé, mely által újszerű alkalmazásokat és nagy teljesítményt igénylő feladatok megoldását támogatja.

A grid lényegesen különbözik a többi, hagyományos elosztott rendszertől, melyek az erőforrások birtoklásán és nem megosztásán alapulnak, ezért használata újszerű megoldásokat igényel a biztonság, a rendszerinformáció, az erőforrások felkutatása, igénylése, ütemezése, folyamatok megfigyelése és irányítása, hibatűrés, naplózás és sok egyéb tekintetében.

Ebben a projektben a grid alkalmazások és erőforrások teljesítményanalízisével kapcsolatban végeztünk kutatást. Workflow alkalmazások (mint a grid egyik tipikus alkalmazás-osztálya) hatékony végrehajtását vettük alapul, és ennek kapcsán próbáltunk újszerű megoldásokat találni a teljesítményanalízis és teljesítmény optimalizáció területén. A teljesítményanalízis önmagában nem ad teljes megoldást, ezért szükséges volt az ehhez kötődő egyéb technikák vizsgálata, amelyek grid alkalmazások fejlesztéséhez és futtatásához (egyebek mellett felhasználói interfész, transzparencia biztosítása, brókerek alkalmazása) illetve grid környezetben való adaptációjához (hibatűrés, migráció, teljesítménykontroll, naplózás) kapcsolódnak.

Ennek megfelelően a kutatási tevékenység az alábbi témák köré csoportosítható:

1. Grid monitorozás és teljesítményanalízis
2. P-GRADE workflow modell és portál
3. Naplózás (checkpoint) technika továbbfejlesztése

## **Grid monitorozás és teljesítményanalízis**

A teljesítményanalízis több, jellegében különböző tevékenységet foglal magában. A fizikailag létező számítási infrastruktúra megfigyelésével, monitorozásával bizonyos adatokat lehet nyerni. Ezen adatok igen sokfélék és nagymennyiségűek, ezért nem hordoznak közvetlenül feldolgozható információt. A nyers adatokat megfelelő mérőszámokká kell alakítani, amelyek jellemezni tudják az alkalmazás és az

infrastruktúra együttes működését. A kiválasztott mérőszámok vizsgálatával következtetni lehet az alkalmazás teljesítményére, illetve az esetleges teljesítményproblémákra. Ez egyfelől megjelenítható emberi vizsgálódásra is alkalmas grafikus formában, illetve közvetlen visszacsatolással szolgálhat az infrastruktúra vagy az alkalmazás felé. Az elvégzett munka az alábbi nagyobb csoportokra bontható:

- **A számítóhálózaton felmerülő lehetséges teljesítményproblémák felmérése.** Több publikáció született, amelyek összegyűjtik mindazokat a teljesítményanalízissel kapcsolatos problémákat, amelyek a grid szemantikai újszerűségéből adódnak [22][55].
- **A teljesítményre jellemző paraméterek, mérőszámok kiválasztása.** A teljesítményjellemzők három célt szolgálhatnak: a végrehajtás módjának leírása, az esetleges teljesítményproblémák kimutatása és azoknak csökkentésére szolgáló módosításokra utalás. Olyan mérőszámokat kell találni, amelyek heterogén környezetben is összehasonlítható módon jellemzik az erőforrásokat és a folyamatokat, illetve világosan elkülöníteni azokat a jellemzőket, amelyeket az infrastruktúra jellegéből adódóan nem lehet egzaktul leírni. Erre vonatkozóan született egy javaslat, mely benchmark programok statisztikáit szimbolikus feldolgozással egyesíti [12].
- **Monitor eszközök fejlesztése.** A grid erőforrásainak és a végrehajtott programok szabályos vagy rendellenes működésének megállapításához szükséges azokról információt gyűjteni. Az információ kezelhetetlenül nagy mennyisége pedig szükségessé teszi azok megfelelő megjelenítését.
- A Gridlab projekt keretében fejlesztett Mercury [7], amely a grid teljesítményanalízis speciális követelményeit figyelembe véve készült. A szokásos kritériumokon túlmenően támogatást nyújt az erőforrás és alkalmazás együttes megfigyeléséhez, a különböző szintű metrikák definiálásához és szétválasztásához, aktív teljesítménykontrollhoz, az adatok előfeldolgozásához.
- Párhuzamos programok teljesítmény monitorozása és nagy mennyiségű adat átvitele a hálózaton, mely a GRM monitor fejlesztésén keresztül valósult meg. A GRM összekötésre került a relációs adatmodell alapú grid információs rendszerrel, az EU-DataGrid projektben fejlesztett R-GMA-val [11].
- A GRM monitor párhuzamos programok teljesítmény monitorozására készült, melyet integráltuk a Mercury monitorozó rendszerrel a hatékonyabb adatátvitel érdekében [19]. Ezzel lehetővé tettük, hogy különféle számítóháló rendszereken (magyar KlaszterGrid, magyar SzuperGrid, JGrid [6][23], európai EGEE infrastruktúra) egységes alkalmazás-megfigyelő rendszerünk legyen.
- a fizikailag létező számítási infrastruktúra és a programok megfigyelése, az adatgyűjtés előfeltétele az optimalizációnak, de önmagában nem nyújt rá megoldást. Ezért a Mercury-t kiterjesztettük az ún. *actuator* képességekkel, amellyel a Mercury-n keresztül vissza lehet hatni a megfigyelt alkalmazásokra és esetleg az erőforrásokra is, ha ezt lehetővé teszik [49].

- A grid-en futó alkalmazások hibakeresésének (debugging) lehetővé tételéhez a Mercury-t kiterjesztettük a távoli hibakeresés támogatásával. Ez az előző szakaszban megvalósított alkalmazás szintű metrikák definiálásának lehetőségére és az ún. *actuator* képességekre épül. Egy alkalmazás specifikus metrika és egy control (amit egy távoli hibakeresést támogató actuator-ban implementáltunk) hozzáadásával lehetővé tettük, hogy a távoli alkalmazás folyamathoz kapcsolódó hibakereső szolgáltatás és a felhasználó gépén futó hibakereső alkalmazás egy virtuális terminál kapcsolat segítségével kommunikáljon a Mercury-n keresztül. Ez lehetővé teszi a felhasználó számára, hogy a megszokott környezetében végezze a grid-en futó alkalmazások hibakeresését, ugyanakkor nem jelent többlet erőforrás felhasználást ha nem használja ezt a lehetőséget [46].
- Az MPI szabványnak megfelelő API-t használó alkalmazások teljesítmény monitorozásához kifejlesztettünk egy ún. *wrapper* könyvtárat, ami az alkalmazás automatikus instrumentálását (a monitorozáshoz szükséges kiegészítések programkódba illesztését) teszi egyszerűbbé. Az alkalmazás instrumentálásához így elegendő a programot a wrapper könyvtárral szerkeszteni, más módosításra nincs szükség. A wrapper könyvtár prototípusához a szabadon elérhető MPICH MPI implementációt vettük alapul, de mivel a wrapper könyvtár is automatikusan generálódik más implementációkhoz is könnyen adaptálható. A wrapper könyvtár a Mercury/GRM alapú grid alkalmazás monitor infrastruktúrával integrált rendszert alkot, ezért az így instrumentált alkalmazások teljesítmény analízise és nyomkövetése a korábban kifejlesztett Prove vizualizációs eszközzel lehetséges [46].
- Az alkalmazások teljesítmény analízise és megfigyelése során előforduló nagy mennyiségű teljesítmény monitorozási esemény kezeléséhez szükség van a monitorozás mélységének dinamikus szabályozására. Ehhez egy új Esemény Kezelő komponenst terveztünk és implementáltunk a Mercury-hoz, amely lehetővé teszi a teljesítmény monitorozási események előfeldolgozását, és a felhasználó által meghatározott feltételeknek megfelelően, magasszintű eseményekké alakítását. Az Esemény Kezelő lehetővé teszi a felhasználónak, hogy dinamikusan, az alkalmazás futása közben határozhassa meg és módosíthassa a teljesítmény analízis igényei szerint a magasszintű eseményeket.
- A CoreGrid projekt keretében javaslatot tettünk egy többszintű monitor architektúrára, amely lehetővé teszi több, már meglevő, de egymással nem kompatibilis monitor rendszer egyidejű használatát. A felső szinten egy leírónyelvvvel lehet megadni, hogy az egyes termelők és fogyasztók milyen paraméterekkel rendelkeznek (beszért protokoll, kért ill. szolgáltatott adatok stb.). Ezen leírások alapján egy később kidolgozandó komponens képes a termelőket és fogyasztókat egymáshoz rendelni, esetleg több lépcsőn keresztül, ha pl. a termelő és a fogyasztó nem beszél közös protokollt, és emiatt konverzióra van szükség.

- **Vizualizációs módszerek és eszközök fejlesztése.** A teljesítményanalízis közvetlen visszajelzéssel szolgálhat az alkalmazás (migráció) vagy az infrastruktúra (bróker) felé. Másfelől azonban szükséges a folyamatok emberi felügyelete is, amely a nyert információ mennyisége és komplexitása miatt csak valamilyen intelligens grafikus, a grid erőforrások és job adatok, az információ forrásától és az adatok szintaxisától független feldolgozásával és megjelenítésével lehetséges [4].
- A **Pulse** eszköz egy Javában fejlesztett program, amely elválasztja az adatforrást, az elemzést és a grafikus prezentációt egymástól és így rugalmasan adaptálható a különböző információ megjelenítési igényekhez [8].
- A GRM-nek eredetileg része a **Prove** vizualizációs eszköz. Ennek fejlettebb verziója készült el, amely nemcsak önálló alkalmazásként, hanem Java appletként illetve portálba beépítve is képes működni [24]. Az új Prove legfontosabb tulajdonsága, hogy képes távolban elkészült adatfájlok kezelésére is, lehetővé téve ezzel a számítógépen való alkalmazását. Emellett workflow vizualizációját is támogatja, azaz egy felsőbb szintű nézetben az egész program-gráf állapotát mutatja, a részletekbe merülve pedig egy-egy (párhuzamos) program teljesítményét lehet megfigyelni.
- Mindezek mellett az APART-2 Network of Excellence projekt keretében a különböző Grid monitorozás, teljesítményanalízis és vizualizáció területén kutatásokat végző partnerekkel elkészítettünk egy alapos képesség-összehasonlító ún. white paper-t a fellelhető összes ilyen eszközről. Ez a dokumentum egyrészt a létező eszközök tárja, másrészt az első lépés abban az irányban, hogy a különböző kutatásokat összehangoljuk, megegyezzünk a közös terminológiában és definiáljuk a terület egyes feladatait és megoldási lehetőségeit [33].

## P-GRADE workflow modell és portál

Kidolgoztuk a P-GRADE rendszer [9][10][15] egy újabb absztrakciós szintjét, az ún. workflow réteget [13], aminek a segítségével a felhasználó a már korábban elkészített P-GRADE párhuzamos alkalmazások, vagy más típusú (szekvenciális, PVM vagy MPI üzenetközvetítésen alapuló) alkalmazások közötti függőségeket tudja leírni. Így a felhasználó grafikusan meg tudja határozni az egyes alkalmazások lefutási sorrendjét, illetve magukat a kimeneti és bemeneti adatállományokat is. A kutatás eredményeképp, a P-GRADE futtatórendszere képes végrehajtani a workflow segítségével leírt komplex alkalmazásokat Grid környezetben is (kiaknázva a CONDOR/CONDOR-G/DAGman és Globus Toolkit adta lehetőségeket), valamint automatikusan eljuttatni a különböző Grid erőforrások között a be- illetve kimeneti állományokat GridFTP-re alapozva. További eredmény, hogy a kidolgozott rendszer lehetőséget biztosít akár workflow szinten, akár az egyes részfeladatok szintjén is információkat begyűjteni a futásról, és azokat on-line vizualizálni a felhasználó számára teljesítményanalízis céljából.

Ehhez kapcsolódóan elkészült a P-GRADE portál [14], melynek segítségével a felhasználók egy web böngészőn keresztül szerkeszthetnek workflow alkalmazásokat és

hajthatnak végre számítógépen. A portál képes a biztonsági tanúsítványok kezelésére, és a teljesítmény-vizualizációs eredmények hatékony megjelenítésére a kliens oldalon. A kialakított portál a GridSphere technológiára alapul, tehát a kialakított workflow eszköz számos jól bevált technológiát ötvöz össze egyetlen környezetbe, elősegítve a különböző irányzatok konvergenciáját.

- **A fejlesztések eredményeként megszületett egy egységes absztrakciós réteget használó portál**, melynek segítségével a felhasználók egyszerre több (nemcsak földrajzilag, illetve virtuális szervezetbe tartozó, hanem akár különböző technológiákra alapuló) számítógépes infrastruktúrán is képesek workflow-kat futtatni. A kidolgozott rendszer hatékonyságát és jelentőségét jól bizonyítja, hogy a SEE-GRID, illetve a HUNGRID [36][51] virtuális szervezetek saját hivatalos portálnak választották, és többek között a VOCE (Közép-Európai virtuális szervezet), az EGRID, valamint a UK NGS is P-GRADE portálon keresztül biztosít felhasználóinak elérhetőséget a számítógépes infrastruktúrájukhoz.
- **A portál lehetőséget biztosít akár workflow szinten, akár az egyes részfeladatok szintjén is információkat begyűjteni a futásról**, és azokat on-line vizualizálni a felhasználó számára teljesítményanalízis céljából. Ehhez a Mercury/GRM alapú grid alkalmazás monitor infrastruktúrát integráltuk a rendszerbe.
- **A portál egyik legnagyobb előnye, hogy különböző számítógépes implementációkat támogat** [39][54]. Eredetileg az ún. 2. generációs Grid rendszereket, a Globus csomagra alapuló implementációkat támogattuk, de ezek között is számos különbség van. Az ún. 3 generációs, szolgáltatás-alapú számítógépes kutatások standardja az OGSA (Open Grid Service Architecture). A portált sikeresen alakítottuk át a brit fejlesztésű „UK OGSA test-bed”-hez, bizonyítva, hogy a portál az eredetitől teljesen eltérő szemléletű számítógépes koncepciókhoz is jól alkalmazható és gyorsan illeszthető [29][32].
- **A számítógépes infrastruktúrán az erőforrások jobb kihasználtságáért az úgynevezett erőforrás brókerek felelősek**. Kutatási területünket kiterjesztettük, a portál ezen brókerekkel való hatékony együttműködésének vizsgálatára is. Munkánk során többféle erőforrás brókerhez sikeresen illesztettük a portált. Az illesztéseknek köszönhetően a felhasználónak nem szükséges pontos információkkal rendelkeznie a számítógépes hálón aktuálisan működő erőforrásokról, és így könnyebben tud workflow-kat futtatni az adott infrastruktúrán [37].
- További feladatként a számítógépes dinamikus viselkedésének ellensúlyozására kerestünk megoldási alternatívákat. **A portál hibátűrő képességének elemzését követően, felkészítettük a portált a számítógépes hibájából bekövetkező sikertelen workflow futások eseményvezérelt javítására**. A megvalósított módosítások hatására, a workflow-k futása közben bekövetkező esetleges hibákat, a felhasználó könnyedén kijavíthatja anélkül, hogy a teljes workflow-t újra kellene futtatnia a számítógépes hálón. A fejlesztés folytatásaként a workflow-k könnyebb felügyeletéhez és kezeléséhez olyan új modulok kerültek kifejlesztésre a portálban, melyek workflow méretkorlátozást, illetve az elkészített workflow-k importálását és exportálását teszik lehetővé.

- **A kifejlesztett portál teljesítményanalízise elengedhetetlen feltétele volt a rendszer működésének megfelelő mélységű vizsgálatához.** A portálon teszteléseket végeztünk mind teherbírás, mind késleltetések, mind megbízhatóság, mind pedig hibátűrés szempontjából. A teszteredmények ismeretében meghatároztuk az optimális működéshez szükséges működési paramétereket, melyek jelentős erőforrásigény csökkenést, és ennek megfelelően megnövekedett teherbírást eredményeztek. A tesztelések eredményei, illetve a kifejlesztett automatizált tesztelési megoldások nemcsak a portál, hanem az alatta működő számítóháló teljesítményanalíziséhez is hatékonyan felhasználhatók lehetnek az elkövetkezőkben.
- **Végezetül a kollaboratív workflow tervezés és végrehajtás területén végzet kutatásaink eredményeként elkészült a kollaboratív portál prototípusa,** melyben több, egymástól földrajzilag távol lévő személy képes ugyanazon workflow alkalmazáson dolgozni [38]. Az angliai Reading-i Egyetem munkatársaival együttműködve kialakított prototípus képes a különböző Virtuális szervezetekhez (VO) tartozó felhasználók által létrehozott workflow alkalmazást a számítóhálón úgy végrehajtani, hogy közben feloldja a résztvevő VO-k összeszervezésével, a szükséges engedélyek megfelelő időben történő beszerzésével és az adatfájlok különböző VO-k közötti mozgatásával kapcsolatos problémákat [40][41].
- **A P-GRADE rendszert sikeresen alkalmaztuk egy kémiai reakció-diffúziós szimuláció párhuzamosítására [16].** Az elkészült rendszer igen jól skálázható akár 50 processzorig, és jó alapot biztosít ahhoz, hogy a már kialakított workflow támogatás segítségével integrálhassuk más kémiai [43] és meteorológiai [34][53] alkalmazásokkal, így elősegítve a hatékonyabb szmogriadó-tervek készítését a közeljövőben, illetve a felszín közeli ózon mennyiségének pontosabb meghatározását.

## Checkpoint technika továbbfejlesztése

Az alkalmazások végrehajtását csak úgy lehet optimalizálni, ha a bróker hozhat olyan döntéseket, hogy az alkalmazás egyes részeit, vagy az egészet áthelyezi más erőforrásokra. Az ezt biztosító migrációs képesség alapja a megfelelő checkpoint (ellenőrzőpont-készítő, naplózó) rendszer kidolgozása. Kezdetben a P-GRADE rendszerben kifejlesztett (GRAPNEL) párhuzamos programok naplózását oldottuk meg, majd általánosítottuk szabványos PVM illetve MPI alkalmazások naplózására [21]. Ez a technika elengedhetetlen a párhuzamos alkalmazások hibátűrésének támogatásához is [20].

- **Kiterjesztettük a P-GRADE által generált párhuzamos alkalmazások naplózhatóságát** (checkpoint) csoportokat és template-eket használó alkalmazások számára is. Ezen alkalmazások multi-pont típusú kommunikációját kellett felkészíteni a megszakíthatóságra. Mivel az ilyen típusú kommunikációkat a PVM rendszerben egy külső szerver modul végzi és ennek módosítására, illetve befolyásolására nincs lehetőség, ezért helyettesítettük e szerver funkcionalitását GRAPNEL szinten. A scatter, gather, multicast típusú kommunikációra a már korábban kiépített pont-pont kommunikációt naplózni képes algoritmusunkat alkalmaztuk.

- **Kierjesztettük a PGRADE által generált párhuzamos alkalmazások naplózó rendszerét az alkalmazás által használt fájlokra is.** Az alkalmazás által megnyitott, írt, olvasott fájlok a szolga folyamatok számára folyamatosan elérhetőek függetlenül attól, hogy a szolga folyamat mely gépekre vándorolt el.
- **Hozzáillesztettük a P-GRADE naplózó rendszerét a Condor feladatütemezőjéhez,** abból a célból, hogy előkészítsük az alkalmazások futtathatóságát és naplózhatóságát grid környezetben is. Ilyen esetben a Condor feladatütemező dinamikusan foglal szabad erőforrásokat az alkalmazás futtatásához, illetve túlterheltség esetén elveszi azt az alkalmazástól. Ez utóbbi esetben a Condor jelzésének hatására az alkalmazás állapottere mentésre kerül, majd azok a folyamatok, melyek a kiűritendő gépeken futnak, kilépnek, majd újraindulnak más szabad gépeken. Ezzel a mechanizmussal az alkalmazásba épített naplózó és migrációs rendszer képes az alkalmazást folyamatosan életben tartani egy dinamikusan változó futási környezetben [20].
- Condor alapú grid környezetben a párhuzamos alkalmazások futtatásának egyik leggyakoribb módja a mester/szolga típusú topológia. Az első kitüntetett folyamat az alkalmazás teljes ideje alatt fut a kapcsolattartó (submit) gépen, mely folyamatosan rendelkezésre áll. A mester folyamat által indított további szolga folyamatok azonban olyan végrehajtó gépre kerülnek, melyek az alkalmazás élete folyamán kiléphetnek a futtató gépek köréből. Tehát a Condor alapú futtatásnál a mester folyamat folyamatosan életben van. Kiterjesztettük a párhuzamos alkalmazások naplózását a mester folyamatra is. Bár e folyamat vezérli a számítást végző folyamatok lementését, az összes folyamat kilépésekor mégis képesnek kell lennie lementenie magát, a grid egy teljesen új szegmensébe való átköltözés vagy migráció céljából [5][3][21][18][28]. A teljes alkalmazás lementése végén checkpoint fájlok reprezentálják a folyamatok és az alkalmazás állapotát. E fájlokat a megfelelő futtató környezetbe áthelyezve, némi előkészületet követően az alkalmazás képes újraindulni abból az időpontból, amikor a lementés megtörtént. Ezáltal az alkalmazás képes teljes mértékben, minden egyes folyamatával együtt vándorolni, ellentétben a korábbi megoldással, melyben a mester folyamat helyhez kötött volt.
- **Sikeresen illesztettünk a rendszerhez egy automatikus terhelés elosztó/kiegyenlítő modult** is, mely képes az alkalmazást futtató gépek terheltségétől függően vezérelni a naplózó rendszert és így áthelyezni a szolga folyamatokat a hatékonyabb és gyorsabb végrehajtás reményében, ahogy azt valós alkalmazásokon is teszteltük [30].
- **Továbbfejlesztettük a P-GRADE által generált párhuzamos PVM alkalmazásokban kidolgozott módszereket** oly módon, hogy egy P-GRADE fejlesztői környezet nélkül készített (azaz nem GRAPNEL specifikus, a GRAPNEL programok sajátosságait és a belső felépítésből és működéséből adódó támogatást mellőző) PVM vagy MPI alkalmazás is naplózható legyen [47].
- Egy általános, megszorításokat nem tartalmazó párhuzamos program naplózásának megvalósításához szükség van újabb módszerek kidolgozására.



Ezért egy új, PVM naplózást megvalósítani képest eszközt fejlesztettünk ki, melynek neve TotalCheckpoint [48]. Sajátossága, hogy egy PVM alkalmazás naplózását a felhasználói kód módosítása nélkül képes megvalósítani, a koordinációt egy külső háttérben futó segédfolyamat végzi. A PVM alkalmazás naplózását kiegészítettük az alkalmazás által használt fájlok állapotainak lementésével és visszaállításával.

- Míg a P-GRADE esetében az alkalmazás része volt a koordinátor, a TotalCheckpoint rendszerben ez egy külső modul. Ennek áthidalására dolgoztunk ki egy újabb módszert melynek célja a koordinációnak az alkalmazás részévé tétele. Ezzel sikerült elérni, hogy az alkalmazás önmaga állapotát lementeni és visszaállítani képes legyen oly módon, hogy a programozónak nem kell speciális programozást követnie, a kód változatlan maradhat. További előnye ennek a módszernek, hogy a futtató környezetet sem kell módosítani annak érdekében, hogy a naplózás miatt az alkalmazás működésében bekövetkező változásokhoz illeszkedjen, mert ezek a változások a külső eszközök számára nem érzékelhetők.
- A TotalCheckpoint rendszer fejlesztésének folytatásaként a lengyel Poznani Szuperszámítógép és Hálózati Központ munkatársaival azon dolgozunk, hogy a párhuzamos naplózó rendszer azon komponensét, mely egyetlen folyamat lementéséért felel, szabványos interfésszel lássuk el. Ennek következményeként a TotalCheckpoint rendszerünk mindig az éppen használt infrastruktúrán megtalálható leginkább optimalizált és legjobb teljesítményt produkáló helyileg elérhető szekvenciális naplózót használja, így egyrészt a lementés és visszaállítás teljesítménye is javul, továbbá a naplózó önmagában is lehetővé teszi egy adott infrastruktúra jobb kihasználását és ezáltal a teljesítményének növelését.

Budapest, 2006. február 23.

Kacsuk Péter  
projektvezető

## Hivatkozások

1. Kovács J., Kacsuk P.: The DIWIDE Distributed Debugger Parallel and Distributed Computing Practices. Special Issue: Quality of Parallel and Distributed Programs and Systems (4), 4, pp. 5-24. 2003
2. Podhorszki N., Kacsuk P: Semi-on-line monitoring of P-GRADE Applications Parallel and Distributed Computing Practices. Special Issue: Quality of Parallel and Distributed Programs and Systems (4), 4, pp. 43-60, 2003
3. Kacsuk P., Lovas R., Kovács J., Szalai F., Gombás G., Podhorszki N., Horváth A., Horányi A., Szeberényi I., Delaitre T., Terstyánszky G., Gourgoulis A.: Demonstration of P-GRADE job-mode for the Grid Euro-Par 2003 Parallel Processing, Lecture Notes in Computer Science, (2790), Klagenfurt, Austria 2003, pp. 1281-1286
4. Podhorszki N., Kacsuk P: Presentation and Analysis of Grid Performance Data EuroPar'2003 Parallel Processing, Lecture Notes in Computer Science, (2790), Klagenfurt, Austria, 2003, pp. 119-126
5. Lovas R., Kovács J., Gombás G., Podhorszki N., Balaton Z., Kacsuk P., Szeberényi I., Delaitre T., Gourgoulis A.: Migration and Monitoring of P-GRADE Parallel Jobs in the Grid. IEEE International Conference on Cluster Computing. Hong Kong, 2003. pp. 8-11.
6. Sipos G., Kacsuk P.: Executing and Monitoring PVM Programs in Computational Grids with Jini Euro PVM/MPI Conference, Venice, Italy Venice, Italy 2003 (LNCS 2840), pp. 570-576.
7. Balaton Z., Gombás, G.: Resource and Job Monitoring in the Grid. 9th International Euro-Par 2003 Parallel Processing, Lecture Notes in Computer Science, Klagenfurt, Austria, (2790), 2003. pp. 404-411.
8. Podhorszki N.: Pulse: A Tool for Presentation and Analysis of Grid Performance Data MIPRO'2003, 26th International Conference on Hypermedia and Grid Systems, Opatija, Croatia, 2003 pp. 129-134.
9. Kacsuk P., Dózsa G.: From Supercomputing Programming to Grid Programming by P-GRADE. WESIC'2003, Lillafüred, 2003, pp. 483-494. Invited paper
10. Kacsuk P.: Development and Execution of HPC Applications on Clusters and Grid by P-GRADE. 2003 European Simulation and Modelling Conference, Naples, Italy, 2003. pp. 6-13.
11. Podhorszki N., Kacsuk P.: Monitoring Message Passing Applications in the Grid with GRM and R-GMA EuroPVM/MPI'2003, Lecture Notes in Computer Science 2840, Venice, Italy, 2003, pp. 603-610.
12. Németh Zs., Grid performance, grid benchmarks, grid metrics. 3rd Cracow Grid Workshop, Cracow, 2003 Invited paper
13. Lovas R., Dózsa G., P. Kacsuk, N. Podhorszki, D. Drótos: Workflow Support for Complex Grid Applications: Integrated and Portal Solutions 2nd European Across Grids Conference, Nicosia, Cyprus, 2004 LNCS 3165/2004, pp. 129-138
14. Németh Cs., Dózsa G., Lovas R., Kacsuk P.: The P-GRADE Grid portal, The 2004 International Conference On Computational Science and its Applications, Proc. of ICCSA, Assisi, Italy, 2004

15. Kacsuk P., Dózsa G., Kovács J., Lovas R., Podhorszki N. Balaton Z., Gombás G.: P-GRADE: a Grid Programming Environment Journal of Grid Computing, Vol.1, no. 2, 2004, pp. 171-197
16. Lovas R., Kacsuk P., Lagzi I., Turányi T.: Unified development solution for cluster and grid computing and its application in chemistry. The 2004 International Conference on Computational Science and its Applications, Assisi, Italy, 2004
17. Hermann G., Czifrus Sz.: Case Study of the Easy Parallelisation of a Monte-Carlo Simulation Program with the P-GRADE Development Tool. microCAD International Scientific Conference, University of Miskolc, Miskolc March 18-19, 2004 p.165-169
18. Kovács J., Kacsuk P.: A migration framework for executing parallel programs in the Grid. 2nd European Across Grids Conference, Nicosia, Cyprus, Jan. 28-30, 2004. LNCS 3165/2004, pp. 80-89.
19. Podhorszki N. Balaton Z., Gombás G.: Monitoring Message Passing Parallel Applications in the Grid with GRM and Mercury Monitor. 2nd European Across Grids Conference, Nicosia, Cyprus, 2004 January 28-30, 2004. LNCS 3165/2004, pp. 179-181.
20. Kovács J., Kacsuk P.: Improving fault-tolerant execution for parallel applications under Condor., microCAD International Scientific Conference, University of Miskolc, March 18-19, 2004, pp. 251-256
21. Kovács J., Kacsuk P.: Párhuzamos programok vándorlása a Grid-en., Miskolci Egyetem, Doktoranduszok fóruma, Gépészmérnöki kar szekciókiadványa, 2003, pp.158-164
22. Németh Zs., Gombás G., Balaton Z.: Performance Evaluation on Grids: Directions, Issues, and Open Problems. Proceedings of the Euromicro Conference on Parallel, Distributed and Network-based Processing PDP 2004, A Coruna, Spain, IEEE Computer Society Press, pp. 290-297.
23. Sipos G., Kacsuk P.: Connecting Condor Pools into Computational Grids by Jini Second European AcrossGrids Conference, AxGrids 2004, Nicosia, Cyprus, 2004, pp. 110-113.
24. Sipos G., Kacsuk P.: PROVE Trace Visualisation Tool as a Grid Service Computational Science and Its Application (ICCSA) 2004 Conference, Assisi Italy. Part II, pp. 37-45.
25. Sipos G., Kacsuk P.: Using Jini to Connect Condor Pools into a Computational Grid. MIPRO Conference, Hypermedia and Grid Systems session, Opatija, Croatia, 2004. pp. 197-202.
26. Hermann G.: Linear speedup of a Monte-Carlo Simulation Program With the P-GRADE Development Tool. Proc. on CD on the Networkshop 2004 Győr, <http://nws.iif.hu/ncd2004/index.htm>
27. Patvarczki J., Dózsa G., Kacsuk P.: The Hungarian Supercomputing Grid in the actual practice. 27th International Convention Mipro Opatija, Adriatic Coast, pp. 203-207. Croatia 2004

28. Kovács J.: Process Migration in Clusters and Cluster Grids Distributed and Parallel Systems: Cluster and Grid Computing, Kluwer International Series in engineering and Computer Science, Vol. 777, Dapsys 2004, Budapest, Hungary, pp. 103-110.
29. P. Kacsuk, A. Goyeneche, T. Delaittre, T. Kiss, Z. Farkas, T. Boczko: High-level Grid Application Environment to Use Legacy Codes as OGSA Grid Services. Fifth IEEE/ACM International Workshop on Grid Computing, Pittsburgh, Pennsylvania, USA, 2004, pp. 428-435.
30. Lagzi I., Lovas R., Turányi T.: Development of a Grid Enabled Chemistry Application Distributed and Parallel Systems: Cluster and Grid Computing, Kluwer Intl. Series in Engineering and Computer Science, Vol. 777 (DAPSYS 2004), Budapest, Hungary, pp. 137- 144
31. Lovas R., Vécsei B.: Integration of formal verification and debugging methods in P-GRADE environment Distributed and Parallel Systems: Cluster and Grid Computing, Kluwer Intl. Series in Engineering and Computer Science, Vol. 777 (DAPSYS 2004), Budapest, Hungary, pp. 83-92
32. P. Kacsuk, A. Goyeneche, T. Delaittre, T. Kiss, Z. Farkas, T. Boczko: High-Level Grid Application Environment to Use Legacy Codes as OGSA Grid Services ERCIM News No. 59, October 2004
33. Gerndt M., Wismüller R., Balaton Z., Gombás G., Kacsuk P., Németh Zs., Podhorszki N., Truong H.L., Fahringer T., Bubak M., Laure E., Margalef T.: Performance Tools for the Grid: State of the Art and Future LRR-TUM Research Report Series, Volume 30, Aachen 2004. ISBN:\* 3832224130
34. Lovas R., Kacsuk P., Horváth A., Horányi A.: Application of P-GRADE Development Environment in Meteorology. In: Distributed and Parallel Systems. Special issue of Scalable Computing: Practice and Experience. Electronic Journal, <http://www.scpe.org>, Vol 6, No 2: ISSN 1895-1767. 2005.
35. Lusthaus Z., Hermann G., Györkei G., Fodor G.: Parallel Genetic Algorithm Programming and its Grid execution by the P-GRADE portal. In: MIPRO 2005, XXVIII. International Convention. Opatija, Croatia, 2005. pp: 234-239
36. Patvarczki J., Debreczeni G., Lovas R., Lagzi I., Kacsuk P., Turányi T.: A HunGrid bemutatása és alkalmazása levegőszennyezés előrejelzésére, In: NETWORKSHOP 2005, Szeged, Szegedi Tudományegyetem, Tanulmányi Információs Központ. p.63, English p. 138 <http://nws.iif.hu/ncd2005/doCS/ehu/115.pdf>
37. Sipos G., Patvarczki J., Hermann G., Drótos D., Kacsuk P., Farkas Z.: A ClusterGrid és a P-GRADE Portál összekötése. In: Networkshop 2005, Szeged, Szegedi Tudományegyetem, Tanulmányi Információs Központ (TIK), 30/03/2005 – 01/04/2005, p. 62 English p. 137.
38. Sipos G., Kacsuk P.: Collaborative Workflow Editing in the P-GRADE Portal. In: MicroCAD'2005 International computer science conference. Miskolc, 2005. Miskolc, Miskolci Egyetem, 2005. pp. 353-358.
39. Sipos G., Németh Cs., Boczkó T., Kacsuk P.: Providing a Multi-Grid Access Mechanism by the Grade Portal, Proc. of microCAD 2005 In: MicroCAD'2005 International computer science conference. Miskolc, 2005. Miskolc, Miskolci Egyetem, 2005, pp. 359-365.

40. Sipos G., Lewis G.J., Kacsuk P., Alexandrov V.N.: Workflow-oriented Collaborative Grid Portals. Advances in Grid Computing. In: European Grid Conference, EGC 2005. Amsterdam, The Netherlands, 2005, pp. 434-443.
41. Sipos G., Németh Cs., Lewis G.J., Alexandrov V.N., Kacsuk P.: Executing Workflow-Based Grid Applications with the Collaborative P-GRADE Portal. In: UK e-Science All Hands Meeting, Nottingham, UK, 2005. CD.
42. Sipos G., Kacsuk P.: Classification and Implementations of Workflow-Oriented Grid Portals. In: The 2005 International Conference on High Performance Computing and Communications, HPCC-05. Sorrento, Italy, 2005. pp. 684-693.
43. Lovas R., Patvarczki J., Kacsuk P., Lagzi I., Turányi T., Kullmann L., Haszpra L., Mészáros R., Horányi A., Bencsura Á., Lendvay Gy.: Air pollution forecast on the HUNGRID infrastructure. In: Parallel Computing 2005. Malaga, Spain. 2005.
44. Boczkó T., Patvarczki J., Kacsuk P.: Connecting the P-GRADE Portal with the MDS-2 Grid information system. In: MicroCAD'2005 International computer science conference. Miskolc, 2005. Miskolc, Miskolci Egyetem, 2005, pp. 25-31.
45. Sipos G., Kacsuk P.: Service Grids for Smart Organisations. In: 5<sup>th</sup> International Conference of PhD Students. Miskolc . University of Miskolc, Engineering Sciences II, Miskolc, Hungary, 2005. pp. 177-182,
46. Gombás G., Marosi A.CS., Balaton Z.: Grid application monitoring and debugging using the mercury monitoring system. In: Advances in Grid Computing - EGC 2005, volume 3470 of Lecture Notes in Computer Science, pp. 193-199. 2005.
47. Kovács J.: Making PVM applications checkpointable for the Grid. In: MicroCAD'2005 International computer science conference. Miskolc, 2005. Miskolc, Miskolci Egyetem, 2005, pp. 223-228.
48. Kovács J., Farkas Z., Marosi A.Cs.:Ellenőrzőpont támogatás PVM alkalmazások számára a magyar ClusterGriden. In: NETWORKSHOP 2005, Szeged, Szegedi Tudományegyetem, Tanulmányi Információs Központ <http://nws.iif.hu/ncd2005/doCS/ehu/115.pdf>, p. 137
49. Gombás G.: Grid monitoring. In: 5th International Conference of PhD Students Engineering Sciences II, University of Miskolc, Innovation and Technology Transfer Centre, 2005. pp. 49-54.
50. Kacsuk P., - Podhorszki, N. - Sipos, G.: Skálázható Desktop Grid Rendszer. In: Informatika a felsőoktatásban, IF2005, Debrecen, 2005. Debrecen. CDROM. abstract in proceedings: p. 89
51. Kacsuk P., - Podhorszki, N.: A HunGrid Virtuális Szervezet az Intézményi Erőforrások Országos Összefogására. In: Informatika a felsőoktatásban, IF2005, Debrecen, 2005.CDROM Debrecen. CDROM. Debrecen. CDROM. abstract in proceedings: p. 90.
52. Podhorszki, N. - Vida, G.: Alkalmazói programozási felület SETI-jellegu elosztott programokhoz és végrehajtó rendszer a BOINC infrastruktúrára. In: Networkshop 2005, Szeged. NIIF, 2005. CDROM <http://nws.iif.hu/ncd2005>
53. Lovas R., Lagzi I., Kacsuk P., Turányi T.: Grid alkalmazások: légszennyezés előrejelzési esettanulmány. Informatika Korszerű Technikái Konferencia, Dunaújváros, 2005. november 23. CD.

54. Kacsuk P., Sipos G.: Multi-Grid, Multi-User Workflows in the P-GRADE Grid Portal, Journal of Grid Computing Vol. 3, No. 3
55. Németh Zs., Sunderam V.: Virtualization in Grids: A Semantical Approach. In: J.C. Cunha and O.F. Rana (eds.): Grid Computing: Software Environments and Tools. Springer, 2005